Loader and Urzyczyn are Logically Related

Sylvain Salvati[1], Giulio Manzonetto[2], Mai Gehrke[3], and Henk Barendregt[4]

[1] INRIA Bordeaux — Sud-Ouest, Talence, France.
sylvain.salvati@labri.fr
[2] LIPN, CNRS UMR 7030, Université Paris-Nord, Villetaneuse, France
giulio.manzonetto@lipn.univ-paris13.fr
[3] LIAFA, CNRS, Université Paris-Diderot, Paris, France.
mai.gehrke@liafa.jussieu.fr
[4] Radboud University, Intelligent Systems, Nijmegen, The Netherlands
henk@cs.ru.nl

Abstract. In simply typed λ-calculus with one ground type the following theorem due to Loader holds. (i) Given the full model $\mathcal{F}$ over a finite set, the question whether some element $f \in \mathcal{F}$ is λ-definable is undecidable. In the λ-calculus with intersection types based on countably many atoms, the following is proved by Urzyczyn. (ii) It is undecidable whether a type is inhabited.
Both statements are major results presented in [3]. We show that (i) and (ii) follow from each other in a natural way, by interpreting intersection types as continuous functions logically related to elements of $\mathcal{F}$. From this, and a result by Joly on λ-definability, we get that Urzyczyn's theorem already holds for intersection types with at most two atoms.
Keywords: λ-calculus, λ-definability, inhabitation, undecidability.

Introduction

Consider the simply typed λ-calculus on simple types $\mathbb{T}^0$ with one ground type 0. Recall that a hereditarily finite full model of simply typed λ-calculus is a collection of sets $\mathcal{F} = (\mathcal{F}_A)_{A\in\mathbb{T}^0}$ such that $\mathcal{F}_0 \neq \emptyset$ is finite and $\mathcal{F}_{A\to B} = \mathcal{F}_B^{\mathcal{F}_A}$ (i.e. the set of functions from $\mathcal{F}_A$ to $\mathcal{F}_B$) for all simple types A, B. An element $f \in \mathcal{F}_A$ is λ-*definable* whenever, for some closed λ-term M having type A, we have $[\![M]\!] = f$, where $[\![M]\!]$ denotes the interpretation of M in $\mathcal{F}$. The following question, raised by Plotkin in [7], is known as the Definability Problem:

DP: "Given an element f of any hereditarily finite full model, is f λ-definable?"

A natural restriction considered in the literature [5, 6] is the following:

DP_n: "Given an element f of $\mathcal{F}_n$, is f λ-definable?"

where $\mathcal{F}_n$ (for $n \geq 1$) denotes the unique (up to isomorphism) full model whose ground set $\mathcal{F}_0$ has n elements. Statman's conjecture stating that DP is decidable [9] was refuted by Loader [6], who proved in 1993 (but published in 2001) that DP_n is undecidable for every $n > 6$. Such a result was then strengthened by Joly, who showed in [5] that DP_n is undecidable for all $n > 1$.

Theorem 1. *1.* **(Loader)** *The Definability Problem is undecidable.*
2. **(Loader/Joly)** *DP_n is undecidable for every $n > 6$ (resp. $n > 1$).*

Consider now the λ-calculus endowed with the intersection type system CDV (Coppo-Dezani-Venneri [4]) based on a countable set $\mathbb{A}$ of atomic types. Recall that an intersection type σ is *inhabited* if $\vdash_\wedge M : \sigma$ for some closed λ-term M.

The Inhabitation Problem for this type theory is formulated as follows:

IHP: "Given an intersection type σ, is σ inhabited?"

We will also be interested in the following restriction of IHP:

IHP_n: "Given an intersection type σ with at most n atoms, is σ inhabited?"

In 1999, Urzyczyn [10] proved that IHP is undecidable for suitable intersection types, called "game types" in [3, §17E], and thus for the whole CDV. His idea was to prove that solving IHP for a game type σ is equivalent to winning a suitable "tree game" G. An arbitrary number of atoms may be needed since, in the Turing-reduction, the actual amount of atoms in σ is determined by G.

Theorem 2 (Urzyczyn).

1. *The Inhabitation Problem is undecidable.*
2. *The Inhabitation Problem for game types is undecidable.*

The undecidability of DP and that of IHP are major results presented thoroughly in [3, §4A] and [3, §17E]. In the proof these problems are reduced to well-known undecidable problems (and eventually to the Halting problem). However, the instruments used to achieve these results are very different — the proof by Loader proceeds by reducing DP to the two-letter word rewriting problem, while the proof by Urzyczyn reduces IHP to the emptiness problem for queue automata (through a series of reductions). The fact that these proofs are different is not surprising since the two problems, at first sight, really *look* unrelated.

Our main contribution is to show that DP and IHP are actually Turing-equivalent, by providing a perhaps unexpected link between the two problems. The key ideas behind our constructions are the following. Every intersection $\alpha_1 \wedge \cdots \wedge \alpha_k$ of atoms can be viewed as a set $\{\alpha_1, \ldots, \alpha_k\}$, and every arrow type $\sigma \to \tau$ as a (continuous) step function. Moreover, Urzyczyn's game types follow the structure of simple types. Combining these ingredients we build a continuous model $\mathcal{S} = (\mathcal{S}_A)_{A\in\mathbb{T}^0}$ over a finite set of atomic types, which constitutes a "bridge" between intersection type systems and full models of simply typed λ-calculus. Then, exploiting very natural semantic logical relations, we can study the continuous model, cross the bridge and infer properties of the full model.
Our constructions allow us to obtain the following Turing-reductions (recall that if the problem P_1 is undecidable and $P_1 \leq_T P_2$, then also P_2 is undecidable):

(i) Inhabitation Problem for game types $\leq_T$ Definability Problem,
(ii) Definability Problem $\leq_T$ Inhabitation Problem (cf. [8]),
(iii) $\mathrm{DP}_n \leq_T \mathrm{IHP}_n$ (cf. [8]).

Therefore, by (i) and (ii) we get that the undecidability of DP and IHP follows from each other. Moreover, by (iii) and Theorem 1(2) we conclude that IHP_n is undecidable whenever $n > 1$, which is a new result refining Urzyczyn's one.

$$\begin{array}{ll} \Lambda : & M, N, P ::= x \mid MN \mid \lambda x.M, \text{ where } x \in \mathrm{Var} \\ \mathbb{T}^0 : & A, B, C ::= 0 \mid A \to B \\ \mathbb{T}^{\mathbb{A}}_{\wedge} : & \gamma, \rho, \sigma, \tau ::= \alpha \mid \sigma \to \tau \mid \sigma \wedge \tau, \text{ where } \alpha \in \mathbb{A} \end{array}$$

(a) Sets Λ of λ-terms, $\mathbb{T}^0$ of simple types, $\mathbb{T}^{\mathbb{A}}_{\wedge}$ of intersection types over $\mathbb{A}$.

$$\sigma \leq \sigma \ (\mathrm{refl}) \qquad \sigma \wedge \tau \leq \sigma \ (\mathrm{incl}_L) \qquad \sigma \wedge \tau \leq \tau \ (\mathrm{incl}_R)$$

$$(\sigma \to \tau) \wedge (\sigma \to \tau') \leq \sigma \to (\tau \wedge \tau') \quad (\to_\wedge)$$

$$\frac{\sigma \leq \gamma \quad \gamma \leq \tau}{\sigma \leq \tau} \ (\mathrm{trans}) \qquad \frac{\sigma \leq \tau \quad \sigma \leq \tau'}{\sigma \leq \tau \wedge \tau'} \ (\mathrm{glb}) \qquad \frac{\sigma' \leq \sigma \quad \tau \leq \tau'}{\sigma \to \tau \leq \sigma' \to \tau'} \ (\to)$$

(b) Rules defining the subtyping relation $\leq$ on intersection types $\mathbb{T}^{\mathbb{A}}_{\wedge}$.

$$\frac{}{x_1 : \sigma_1, \ldots, x_n : \sigma_n \vdash_\wedge x_i : \sigma_i} \ (\mathrm{ax}) \qquad \frac{\Gamma \vdash_\wedge M : \tau \to \sigma \quad \Gamma \vdash_\wedge N : \tau}{\Gamma \vdash_\wedge MN : \sigma} \ (\to_E)$$

$$\frac{\Gamma, x : \sigma \vdash_\wedge M : \tau}{\Gamma \vdash_\wedge \lambda x.M : \sigma \to \tau} \ (\to_I) \qquad \frac{\Gamma \vdash_\wedge M : \sigma \quad \Gamma \vdash_\wedge M : \tau}{\Gamma \vdash_\wedge M : \sigma \wedge \tau} \ (\wedge_I) \qquad \frac{\Gamma \vdash_\wedge M : \sigma \quad \sigma \leq \tau}{\Gamma \vdash_\wedge M : \tau} \ (\leq)$$

(c) Rules defining the intersection type system CDV.

Fig. 1: Definition of terms, types, subtyping and derivation rules for CDV. The rules for simply typed λ-calculus are obtained from those in (c) leaving out $(\wedge_I)$ and $(\leq)$.

1 Preliminaries: Some Syntax, Some Semantics

To make this article more self-contained, this section summarizes some definitions and results that we will use later in the paper. Given a set X, we write $\mathcal{P}(X)$ for the set of all subsets of X, and $Y \subseteq_{\mathrm{f}} X$ if Y is a finite subset of X.

1.1 Typed Lambda Calculi

We take untyped λ-calculus for granted together with the notions of closed λ-term, α-conversion, (β-)normal form and strong normalization. We denote by Var the set of variables and by Λ the set of λ-terms. Hereafter, we consider λ-terms up to α-conversion and we adopt Barendregt's variable convention.

We mainly focus on two particular typed λ-calculi (see [3] for more details).

The simply typed λ-calculus *à la Curry* over a single atomic type 0. The set $\mathbb{T}^0$ of *simple types* $A, B, C, \ldots$ is defined in Figure 1(a). *Simple contexts* Δ are partial functions from Var to $\mathbb{T}^0$; we write $\Delta = x_1 : A_1, \ldots, x_n : A_n$ for the function of domain $\{x_1, \ldots, x_n\}$ such that $\Delta(x_i) = A_i$ for i in $[1; n]$. We write $\Delta \vdash M : A$ if M *has type* A *in* Δ, and we say that such an M is *simply typable*.

The intersection type system CDV over an infinite set $\mathbb{A}$ of atomic types. This system was first introduced by Coppo, Dezani and Venneri [4] to characterize strongly normalizable λ-terms. The set $\mathbb{T}^{\mathbb{A}}_{\wedge}$ of *intersection types* is given in Figure 1(a) and it is partially ordered by the subtyping relation $\leq$

defined in Figure 1(b). We denote by $\simeq$ the equivalence generated by $\leq$. As usual, we may write $\bigwedge_{i=1}^{n} \sigma_i \to \tau_i$ for $(\sigma_1 \to \tau_1) \wedge \cdots \wedge (\sigma_n \to \tau_n)$.

Contexts $\Gamma = x_1 : \tau_1, \ldots, x_n : \tau_n$ are handled as in the simply typed case. We write $\Gamma \vdash_\wedge M : \sigma$ if the judgment can be proved using the rules of Figure 1(c).

As a matter of notation, given two sets Y, Z of intersection types, we let $Y^\wedge = \{\sigma_1 \wedge \cdots \wedge \sigma_n \mid \sigma_i \in Y \text{ for } i \in [1;n]\}$ and $Y \to Z = \{\tau \to \sigma \mid \tau \in Y, \sigma \in Z\}$.

We now present some well known properties of CDV. For their proofs, we refer to [4], [3, Thm. 14.1.7] and [3, Thm. 14.1.9] respectively.

Theorem 3. *A λ-term M is typable in CDV iff M is strongly normalizable.*

Theorem 4 (β-soundness). *For all $k \geq 1$, if $\bigwedge_{i=1}^{k} \sigma_i \to \rho_i \leq \gamma_1 \to \gamma_2$ then there is a non-empty subset $K \subseteq [1;k]$ such that $\gamma_1 \leq \bigwedge_{i \in K} \sigma_i$ and $\bigwedge_{i \in K} \rho_i \leq \gamma_2$.*

Theorem 5 (Inversion Lemma). *The following properties hold:*

1. *$\Gamma \vdash_\wedge x : \sigma$ iff $\Gamma(x) \leq \sigma$,*
2. *$\Gamma \vdash_\wedge MN : \sigma$ iff there is ρ such that $\Gamma \vdash_\wedge M : \rho \to \sigma$ and $\Gamma \vdash_\wedge N : \rho$,*
3. *$\Gamma \vdash_\wedge \lambda x.M : \sigma$ iff there is $n \geq 1$ such that $\sigma = \bigwedge_{i=1}^{n} \sigma_i \to \sigma_i'$ for some σ_i, σ_i',*
4. *$\Gamma \vdash_\wedge \lambda x.M : \sigma \to \tau$ iff $\Gamma, x : \sigma \vdash_\wedge M : \tau$.*

1.2 Type Structures Modelling the Simply Typed Lambda Calculus

A *typed applicative structure* $\mathcal{M}$ is a pair $((\mathcal{M}_A)_{A \in \mathbb{T}^0}, \bullet)$ where each $\mathcal{M}_A$ is a structure whose carrier is non-empty, and $\bullet$ is a function that associates to every $d \in \mathcal{M}_{A \to B}$ and every $e \in \mathcal{M}_A$ an element $d \bullet e$ in $\mathcal{M}_B$. From now on, we shall write $d \in \mathcal{M}$ to denote $d \in \mathcal{M}_A$ for some A. We say that $\mathcal{M}$ is: *hereditarily finite* if every $\mathcal{M}_A$ has a finite carrier; *extensional* whenever, for all $A, B \in \mathbb{T}^0$ and $d, d' \in \mathcal{M}_{A \to B}$, we have that $d \bullet e = d' \bullet e$ for every $e \in \mathcal{M}_A$ entails $d = d'$.

A *valuation in* $\mathcal{M}$ is any map $\nu_\mathcal{M}$ from Var to elements of $\mathcal{M}$. A valuation $\nu_\mathcal{M}$ *agrees* with a simple context Δ when $\Delta(x) = A$ implies $\nu_\mathcal{M}(x) \in \mathcal{M}_A$. Given a valuation $\nu_\mathcal{M}$ and an element $d \in \mathcal{M}$, we write $\nu_\mathcal{M}[x := d]$ for the valuation $\nu'_\mathcal{M}$ that coincides with $\nu_\mathcal{M}$, except for x, where $\nu'_\mathcal{M}$ takes the value d. When there is no danger of confusion we may omit the subscript $\mathcal{M}$ and write ν.

A *valuation model* $\mathcal{M}$ is an extensional typed applicative structure such that the clauses below define a total interpretation function $[\![\cdot]\!]^\mathcal{M}_{(\cdot)}$ which maps derivations $\Delta \vdash M : A$ and valuations ν agreeing with Δ to elements of $\mathcal{M}_A$:

- $[\![\Delta \vdash x : A]\!]^\mathcal{M}_\nu = \nu(x)$,
- $[\![\Delta \vdash NP : A]\!]^\mathcal{M}_\nu = [\![\Delta \vdash N : B \to A]\!]^\mathcal{M}_\nu \bullet [\![\Delta \vdash P : B]\!]^\mathcal{M}_\nu$,
- $[\![\Delta \vdash \lambda x.N : A \to B]\!]^\mathcal{M}_\nu \bullet d = [\![\Delta, x : A \vdash N : B]\!]^\mathcal{M}_{\nu[x:=d]}$ for every $d \in \mathcal{M}_A$.

When the derivation (resp. the model) is clear from the context we may simply write $[\![M]\!]^\mathcal{M}_\nu$ (resp. $[\![M]\!]_\nu$). For M closed, we simplify the notation further and write $[\![M]\!]$ since its interpretation is independent from the valuation.

The full model over a set $X \neq \emptyset$, denoted by $\mathrm{Full}(X)$, is the valuation model $((\mathcal{F}_A)_{A \in \mathbb{T}^0}, \bullet)$ where $\bullet$ is functional application, $\mathcal{F}_0 = X$ and $\mathcal{F}_{A \to B} = \mathcal{F}_B^{\mathcal{F}_A}$.

The continuous model over a cpo $(D, \leq)$, written $\mathrm{Cont}(D, \leq)$, is the valuation model $((\mathcal{D}_A, \sqsubseteq_A)_{A \in \mathbb{T}^0}, \bullet)$ such that $\bullet$ is functional application and:

- $\mathcal{D}_0 = D$ and $f \sqsubseteq_0 g$ iff $f \leq g$,
- $\mathcal{D}_{A\to B} = [\mathcal{D}_A \to \mathcal{D}_B]$ consisting of the monotone functions from $\mathcal{D}_A$ to $\mathcal{D}_B$ with the pointwise partially ordering $\sqsubseteq_{A\to B}$.

We will systematically omit the subscript A in $\sqsubseteq_A$ when clear from the context.

Note that both $\mathrm{Full}(X)$ and $\mathrm{Cont}(D, \leq)$ are extensional. Moreover, whenever X (resp. D) is finite $\mathrm{Full}(X)$ (resp. $\mathrm{Cont}(D, \leq)$) is hereditarily finite.

Logical relations have been extensively used in the study of semantic properties of λ-calculus (see [2, §4.5] for a survey). As we will see in Sections 4 and 5 they constitute a powerful tool for relating different valuation models.

Definition 1. *Given two valuation models $\mathcal{M}, \mathcal{N}$, a logical relation $\mathscr{R}$ between $\mathcal{M}$ and $\mathcal{N}$ is a family $\{\mathscr{R}_A\}_{A\in\mathbb{T}^0}$ of binary relations $\mathscr{R}_A \subseteq \mathcal{M}_A \times \mathcal{N}_A$ such that for all $A, B \in \mathbb{T}^0$, $f \in \mathcal{M}_{A\to B}$ and $g \in \mathcal{N}_{A\to B}$ we have:*

$$f \ \mathscr{R}_{A\to B}\ g \ \textit{iff}\ \forall h \in \mathcal{M}_A, h' \in \mathcal{N}_A[h\ \mathscr{R}_A\ h' \Rightarrow f(h)\ \mathscr{R}_B\ g(h')].$$

Given $f \in \mathcal{M}_A$ we define $\mathscr{R}_A(f) = \{g \in \mathcal{N}_A \mid f\ \mathscr{R}_A\ g\}$ and, for $Y \subseteq \mathcal{M}_A$, $\mathscr{R}_A(Y) = \bigcup_{f\in Y} \mathscr{R}_A(f)$. Similarly, for $g \in \mathcal{N}_A$ and $Z \subseteq \mathcal{N}_A$ we have $\mathscr{R}^-_A(g) = \{f \in \mathcal{N}_A \mid f\ \mathscr{R}_A\ g\}$ and $\mathscr{R}^-_A(Z) = \bigcup_{g\in Z} \mathscr{R}^-_A(g)$.

It is well known that a logical relation $\mathscr{R}$ is univocally determined by the value of $\mathscr{R}_0$, and that the fundamental lemma of logical relations holds [2, §4.5].

Lemma 1 (Fundamental Lemma). *Let $\mathscr{R}$ be a logical relation between $\mathcal{M}$ and $\mathcal{N}$ then, for all closed M having simple type A, we have $[\![M]\!]^{\mathcal{M}}\ \mathscr{R}_A\ [\![M]\!]^{\mathcal{N}}$.*

2 Uniform Intersection Types and CDV$^\omega$

A useful approach to prove that a general decision problem is undecidable, is to identify a "sufficiently difficult" fragment of the problem. For instance, Urzyczyn in [10] shows the undecidability of inhabitation for a proper subset $\mathcal{G}$ of intersection types called *game types* in [3, §17E]. Formally, $\mathcal{G} = \mathbb{A} \cup \mathcal{B} \cup \mathcal{C}$ where:

$$\mathcal{A} = \mathbb{A}^\wedge, \mathcal{B} = (\mathcal{A} \to \mathcal{A})^\wedge, \mathcal{C} = (\mathcal{D} \to \mathcal{A})^\wedge \text{ for } \mathcal{D} = \{\sigma \wedge \tau \mid \sigma, \tau \in (\mathcal{B} \to \mathcal{A})\}.$$

(Recall that the notations $Y^\wedge$ and $Y \to Z$ were introduced in Subsection 1.1.) In our case we focus on intersection types that are *uniform* with simple types, in the sense that such intersection types follow the structure of the simple types.

Let us fix an arbitrary set $X \subseteq \mathbb{A}$. We write $\mathbb{T}^X_\wedge$ for the set of intersection types based on X.

Definition 2. *The set $\Xi_X(A)$ of* intersection types uniform with $A \in \mathbb{T}^0$ *is defined by induction on A as follows:*

$$\Xi_X(0) = X^\wedge, \qquad \Xi_X(B \to C) = (\Xi_X(B) \to \Xi_X(C))^\wedge.$$

When there is little danger of confusion, we simply write $\Xi(A)$ for $\Xi_X(A)$.

It turns out that game types are all uniform: $\mathcal{A} \subseteq \Xi_{\mathbb{A}}(0)$, $\mathcal{B} \subseteq \Xi_{\mathbb{A}}(0 \to 0)$ and $\mathcal{D} \subseteq \Xi_{\mathbb{A}}((0 \to 0) \to 0)$ thus $\mathcal{C} \subseteq \Xi_{\mathbb{A}}(((0 \to 0) \to 0) \to 0)$. Therefore the inhabitation problem for uniform intersection types over $\mathbb{A}$ is undecidable too.

Theorem 6 (Urzyczyn revisited). *The problem of deciding whether a type $\sigma \in \bigcup_{A \in \mathbb{T}^0, X \subseteq_f \mathbb{A}} \Xi_X(A)$ is inhabited in CDV is undecidable.*

For technical reasons, that will be clarified in the next section, we need to introduce the system CDV$^\omega$ over $\mathbb{A} \cup \{\omega\}$, a variation of CDV where intersection types are extended by adding a distinguished element ω at ground level.

In this framework, the set $\Xi_{X\cup\{\omega\}}(A)$ of *intersection types with ω uniform with A* will be denoted by $\Xi^\omega_X(A)$, or just $\Xi^\omega(A)$ when X is clear. We write ω_A for the type in $\Xi^\omega(A)$ defined by $\omega_0 = \omega$ and $\omega_{B\to C} = \omega_B \to \omega_C$.

The system CDV$^\omega$ over $\mathbb{T}_\wedge^{\mathbb{A}\cup\{\omega\}}$, whose judgments are denoted by $\Gamma \vdash^\omega_\wedge M : \sigma$, is generated by adding the following rule to the definition of $\leq$ in Figure 1(b):

$$\frac{\sigma \in \Xi^\omega_{\mathbb{A}}(A)}{\sigma \leq \omega_A}\ (\leq_A)$$

Therefore CDV$^\omega$ is different from the usual intersection type systems with ω. By construction, for every $A \in \mathbb{T}^0$, the type ω_A is a maximal element of $\Xi^\omega(A)$. Using [3, Thm. 14A.7], we easily get that the Inversion Lemma (Theorem 5) still works for CDV$^\omega$, while the β-soundness holds in the following restricted form.

Recall that $\simeq$ stands for the equivalence generated by $\leq$.

Theorem 7 (β-soundness for CDV$^\omega$). *Let $k \geq 1$. Suppose $\gamma_1 \to \gamma_2 \not\simeq \omega_A$ for all $A \in \mathbb{T}^0$ and $\bigwedge_{i=1}^k \sigma_i \to \rho_i \leq \gamma_1 \to \gamma_2$, then there is a non-empty subset $K \subseteq [1;k]$ such that $\gamma_1 \leq \bigwedge_{i\in K} \sigma_i$ and $\bigwedge_{i\in K} \rho_i \leq \gamma_2$.*

We now provide some useful properties of uniform intersection types.

Lemma 2. *Let $\sigma \in \Xi^\omega(A)$ and $\tau \in \Xi^\omega(A')$. Then we have that $\sigma \leq \tau$ entails $A = A'$.*

To distinguish arbitrary contexts from contexts containing uniform intersection types (with or without ω) we introduce some terminology.

We say that a context Γ is a *Ξ-context* (resp. *Ξ^ω-context*) if it ranges over uniform intersection types (resp. with ω). A Ξ-context (resp. Ξ^ω-context) $\Gamma = x_1 : \sigma_1, \ldots, x_n : \sigma_n$ is *uniform with* $\Delta = x_1 : A_1, \ldots, x_n : A_n$ if every σ_i belongs to $\Xi(A_i)$ (resp. to $\Xi^\omega(A_i)$).

Lemma 3. *Let $\rho \in \mathbb{T}_\wedge^{\mathbb{A}\cup\{\omega\}}$, $\tau \in \Xi^\omega(B)$ and Γ be a Ξ^ω-context. Then we have that $\Gamma, x : \tau \vdash^\omega_\wedge xN_1 \cdots N_k : \rho$ iff there are $A, A_1, \ldots, A_k \in \mathbb{T}^0$ and $\sigma \in \Xi^\omega(A)$ and $\tau_i \in \Xi^\omega(A_i)$ for i in $[1;k]$ such that $B = A_1 \to \cdots \to A_k \to A$ and:*

1. *$\sigma \leq \rho$,*
2. *$\Gamma, x : \tau \vdash^\omega_\wedge xN_1 \cdots N_k : \sigma$,*
3. *$\tau \leq \tau_1 \to \cdots \to \tau_k \to \sigma$,*
4. *$\Gamma, x : \tau \vdash^\omega_\wedge N_i : \tau_i$ for all i in $[1;k]$.*

Furthermore, if Γ is a Ξ-context, $\rho \in \mathbb{T}_\wedge^{\mathbb{A}}$ and $\tau \in \Xi(B)$, then σ and the τ_i for i in $[1;k]$ may also be chosen as uniform intersection types without ω (while the type judgments $\vdash^\omega_\wedge$ still need to be in CDV$^\omega$*).*

Theorem 8 (Uniform Inversion Lemma for CDV$^\omega$). *Let $\sigma \in \Xi^\omega(A)$ and Γ be a Ξ^ω-context. Then we have that (where we suppose that each term in a type judgment is in normal form):*

1. $\Gamma \vdash^\omega_\wedge x : \sigma$ *iff* $\Gamma(x) \leq \sigma$,
2. $\Gamma \vdash^\omega_\wedge MN : \sigma$ *iff there exist* $B \in \mathbb{T}^0$ *and* $\tau \in \Xi^\omega(B)$ *such that* $\Gamma \vdash^\omega_\wedge M : \tau \to \sigma$ *and* $\Gamma \vdash^\omega_\wedge N : \tau$,
3. $\Gamma \vdash^\omega_\wedge \lambda x.N : \sigma$ *iff* $A = B \to C$ *and there are* $\tau_i \in \Xi^\omega(B), \tau'_i \in \Xi^\omega(C)$ *such that* $\sigma = \bigwedge_{i=1}^n \tau_i \to \tau'_i$ *and* $\Gamma, x : \tau_i \vdash^\omega_\wedge N : \tau'_i$ *for all* i *in* $[1; n]$.

Corollary 1. *For M a normal λ-term, $\sigma \in \Xi^\omega(A)$ and Γ a Ξ^ω-context uniform with Δ, we have that $\Gamma \vdash^\omega_\wedge M : \sigma$ entails $\Delta \vdash M : A$.*

Proof. A simple consequence of the Uniform Inversion Lemma (with Lemma 2 when M is a variable). ☐

The corollary above does not generalize to arbitrary λ-terms as the following example illustrates. Let $M = \lambda zy.y$ and $N = \lambda x.xx$, then we have that $\vdash^\omega_\wedge MN : \alpha \to \alpha \in \Xi^\omega(0 \to 0)$ since $\vdash^\omega_\wedge N : \gamma$ and $\vdash^\omega_\wedge M : \gamma \to \alpha \to \alpha$ where $\gamma = (\beta \wedge (\beta \to \beta)) \to \beta$. However N is not simply typable, hence neither is MN. Note that, while we consider only uniform intersection types, we do not restrict the intersection type systems so that the type γ still may be used in a deduction.

CDV and CDV$^\omega$ are equivalent on normal forms in the following sense.

Lemma 4. *For every normal $M \in \Lambda$, for every Ξ-context $\Gamma = x_1 : \tau_1, \ldots, x_n : \tau_n$ uniform with $\Delta = x_1 : A_1, \ldots, x_n : A_n$, and for every $\sigma \in \Xi(A)$ we have:*

$$\Gamma \vdash_\wedge M : \sigma \iff \Gamma \vdash^\omega_\wedge M : \sigma.$$

Proof. ($\Rightarrow$) Trivial, as CDV is a subsystem of CDV$^\omega$.

($\Leftarrow$) We proceed by induction on the structure of M. The cases where M is a variable or a λ-abstraction can be treated thanks to Theorem 5 for CDV$^\omega$and the induction hypothesis. Concerning the case where $M = x_i N_1 \cdots N_k$, from the ω-free version of Lemma 3, we have that $A_i = B_1 \to \cdots \to B_k \to A$, there exist $\tau_1, \ldots, \tau_k$ respectively in $\Xi(B_1), \ldots, \Xi(B_k)$ such that $\tau_i \leq \tau_1 \to \cdots \to \tau_k \to \sigma$ and $\Gamma \vdash^\omega_\wedge N_i : \tau_i$ for each i in $[1; k]$. Therefore, by the induction hypothesis, we have that for every i in $[1; k]$, $\Gamma \vdash_\wedge N_i : \tau_i$ which entails that $\Gamma \vdash_\wedge M : \sigma$. ☐

3 The Continuous Model over $\mathcal{P}(X)$

Hereafter we consider fixed an arbitrary set $X \subseteq_{\mathrm{f}} \mathbb{A}$. We are going to represent uniform intersection types based on $X \cup \{\omega\}$, as elements of the continuous model $\mathcal{S}$ over $\mathcal{P}(X)$, ordered by set-theoretical inclusion.

Let $\mathcal{S} = \{(\mathcal{S}_A, \sqsubseteq_A)\}_{A \in \mathbb{T}^0} = \mathrm{Cont}(\mathcal{P}(X), \subseteq)$. Each $\mathcal{S}_A$ is a finite join-semilattice and thus a complete lattice. We denote the join by $\sqcup$ and the bottom by $\perp_A$.

Given $f \in \mathcal{S}_A, g \in \mathcal{S}_B$ we write $f \mapsto g$ for the corresponding *step function*:

$$(f \mapsto g)(h) = \begin{cases} g & \text{if } f \sqsubseteq_A h, \\ \perp_B & \text{otherwise.} \end{cases}$$

For all A we define a function $\iota_A : \Xi^\omega(A) \to \mathcal{S}_A$ by induction on A as follows.

Definition 3. *For $\alpha \in X$ and $\sigma, \tau \in \Xi^\omega(0)$ we let $\iota_0(\alpha) = \{\alpha\}$, $\iota_0(\omega) = \bot_0 = \emptyset$, $\iota_0(\sigma \wedge \tau) = \iota_0(\sigma) \sqcup \iota_0(\tau)$. For $\sigma, \tau \in \Xi^\omega(A \to B)$ we define:*

$$\iota_{A\to B}(\sigma \to \tau) = \iota_A(\sigma) \mapsto \iota_B(\tau), \qquad \iota_{A\to B}(\sigma \wedge \tau) = \iota_{A\to B}(\sigma) \sqcup \iota_{A\to B}(\tau).$$

Remark 1. Given $\sigma \in \Xi^\omega(A)$, we have that $\sigma \simeq \omega_A$ entails $\iota_A(\sigma) = \bot_A$.

Thanks to the presence of the maximal element ω_A, the correspondence between $\Xi^\omega(A)$ and $\mathcal{S}_A$ is actually very faithful (in the sense of Corollary 2).

Lemma 5. *Let $h = \bigsqcup_{i=1}^n f_i \mapsto g_i$, then for every f we have:*

(i) $h(f) = \bigsqcup_{i\in K} g_i$ where $K = \{i \in [1;n] \mid f_i \sqsubseteq f\}$.
(ii) $h \sqsubseteq f$ iff $g_i \sqsubseteq f(f_i)$ for all $1 \le i \le n$.

Lemma 6. *Step functions are generators: $\forall f \in \mathcal{S}_{A\to B}$, $f = \bigsqcup_{g\in\mathcal{S}_A} g \mapsto f(g)$.*

Proof. Let $h = \bigsqcup_{g\in\mathcal{S}_A} g \mapsto f(g)$. We need to prove that, for every $g \in \mathcal{S}_A$, $f(g) = h(g)$. From Lemma 5(i), we have that $h(g) = \bigsqcup_{g'\sqsubseteq g} f(g')$. Since f is monotone, we have that for every $g' \sqsubseteq g$, $f(g') \sqsubseteq f(g)$ and therefore $\bigsqcup_{g'\sqsubseteq g} f(g') \sqsubseteq f(g)$. Since obviously $f(g) \sqsubseteq \bigsqcup_{g'\sqsubseteq g} f(g')$, we obtain $f(g) = \bigsqcup_{g'\sqsubseteq g} f(g') = h(g)$. □

Lemma 7. *For all $A \in \mathbb{T}^0$, $\sigma, \tau \in \Xi^\omega(A)$ we have $\sigma \le \tau$ iff $\iota_A(\tau) \sqsubseteq \iota_A(\sigma)$.*

Proof. We proceed by induction on A. In case $A = 0$, the equivalence is clear since $\mathcal{P}(X)$ is the free $\sqcup$-semilattice with bottom over X and $\Xi^\omega(0)/\simeq$ is the free $\wedge$-semilattice with top over X.

In case $A = B \to C$, we have two subcases. Case 1, $\tau \simeq \omega_D$ for some $D \in \mathbb{T}^0$. Then by Lemma 2 we get $D = A$, by Remark 1 we get $\iota_A(\tau) = \bot_A$ and the equivalence follows since both $\sigma \le \tau$ and $\iota_A(\tau) \sqsubseteq \iota_A(\sigma)$ hold. Case 2, $\sigma = \bigwedge_{i=1}^n \sigma_i \to \sigma_i'$, $\tau = \bigwedge_{j=1}^m \tau_j \to \tau_j'$ and $\tau \not\simeq \omega_D$ for any $D \in \mathbb{T}^0$. By Remark 1 we can assume, without loss of generality, that for every j in $[1;m]$ we have $\tau_j \to \tau_j' \not\simeq \omega_D$ for all $D \in \mathbb{T}^0$. (Indeed for those k such that $\tau_k \to \tau_k' \simeq \omega_D$ one reasons as in Case 1.) We now prove the equivalence for this case.

($\Rightarrow$) If $\sigma \le \tau$, then by β-soundness, for every j in $[1;m]$, there is K_j included in $[1;n]$ such that $\tau_j \le \bigwedge_{i\in K_j} \sigma_i$ and $\bigwedge_{i\in K_j} \sigma_i' \le \tau_j'$. By the induction hypothesis:

$$(1) \quad \bigsqcup_{i\in K_j} \iota_B(\sigma_i) \sqsubseteq \iota_B(\tau_j) \qquad\qquad (2) \quad \iota_C(\tau_j') \sqsubseteq \bigsqcup_{i\in K_j} \iota_C(\sigma_i')$$

We now prove that, for every $f \in \mathcal{S}_B$, $\iota_A(\tau)(f) \sqsubseteq \iota_A(\sigma)(f)$. From Lemma 5(i), we get $\iota_A(\tau)(f) = \bigsqcup_{j\in J} \iota_C(\tau_j')$ where $J = \{j \in [1;m] \mid \iota_B(\tau_j) \sqsubseteq f\}$. By definition of J, we have that $\bigsqcup_{j\in J} \iota_B(\tau_j) \sqsubseteq f$ so, by (1), we obtain $\bigsqcup_{j\in J, i\in K_j} \iota_B(\sigma_i) \sqsubseteq f$. Therefore by Lemma 5(i), we get $\bigsqcup_{j\in J, i\in K_j} \iota_C(\sigma_i') \sqsubseteq \iota_A(\sigma)(f)$ and, using (2), we obtain $\iota_A(\tau)(f) \sqsubseteq \iota_A(\sigma)(f)$. As a conclusion we have $\iota_A(\tau) \sqsubseteq \iota_A(\sigma)$.

($\Leftarrow$) If $\iota_A(\tau) \sqsubseteq \iota_A(\sigma)$, then we have in particular $\iota_A(\tau)(\iota_B(\tau_j)) \sqsubseteq \iota_A(\sigma)(\iota_B(\tau_j))$ for each $j \in [1,m]$. From Lemma 5(i), we have that $\iota_A(\tau)(\iota_B(\tau_j)) = \bigsqcup_{i\in I_j} \iota_C(\tau_i')$ where $I_j = \{i \in [1;m] \mid \tau_i \le \tau_j\}$. Since $\tau_j \le \tau_j$ we must have $j \in I_j$ and therefore, we obtain $\iota_C(\tau_j') \sqsubseteq \iota_A(\tau)(\iota_B(\tau_j))$. So, again by Lemma 5(i), we have that

$\iota_A(\sigma)(\iota_B(\tau_j)) = \bigsqcup_{k\in K_j} \iota_C(\sigma'_k)$ where $K_j = \{k \in [1;n] \mid \tau_j \le \sigma_k\}$. Thus we get $\iota_C(\tau'_j) \sqsubseteq \bigsqcup_{k\in K_j} \iota_C(\sigma'_k)$ and hence, by the induction hypothesis, $\bigwedge_{k\in K_j} \sigma'_k \le \tau'_j$. Now, by definition of K_j, we also have $\tau_j \le \bigwedge_{k\in K_j} \sigma_k$. As we can find such a K_j for every j in $[1;m]$, we can finally conclude that $\sigma \le \tau$. □

Corollary 2. *The map ι_A is an order-reversing bijection on $\Xi^\omega(A)/\simeq$.*

Proof. If $\tau \le \sigma$ and $\sigma \le \tau$, then Lemma 7 implies that $\iota_A(\tau) = \iota_A(\sigma)$. From this it ensues that ι_A is an order-reversing injection. To prove that it is actually a bijection, we need to show that ι_A is surjective. We proceed by induction on A. Clearly when $A = 0$, ι_A is surjective. If $A = B \to C$ then we get from the induction hypothesis that ι_B and ι_C are bijections between $\Xi^\omega(B)/\simeq$ and $\mathcal{S}_B$, and between $\Xi^\omega(C)/\simeq$ and $\mathcal{S}_C$, respectively. Now, given f in $\mathcal{S}_A$, we define $\tau_f \in \Xi^\omega(A)$ to be $\bigwedge_{g\in\mathcal{S}_B} \iota_B^{-1}(g) \to \iota_C^{-1}(f(g))$. But, $\iota_{A\to B}(\tau_f) = \bigsqcup_{g\in\mathcal{S}_B} g \mapsto f(g)$ which is equal to f by Lemma 6. □

The above results are related to Stone duality for intersection types (cf. [1]).

Proposition 1. *Let M be a normal term such that $x_1 : A_1, \ldots, x_n : A_n \vdash M : A$. Then for all $\tau_i \in \Xi^\omega(A_i)$, $\sigma \in \Xi^\omega(A)$ the following two sentences are equivalent:*

1. $x_1 : \tau_1, \ldots, x_n : \tau_n \vdash^\omega_\wedge M : \sigma$,
2. $\iota_A(\sigma) \sqsubseteq [\![M]\!]^{\mathcal{S}}_\nu$, *for all valuations ν such that $\nu(x_i) = \iota_{A_i}(\tau_i)$.*

Proof. Let $\Delta = x_1 : A_1, \ldots, x_n : A_n$ and $\Gamma = x_1 : \tau_1, \ldots, x_n : \tau_n$.
($1 \Rightarrow 2$) We proceed by structural induction on M.

- In case $M = x_i$, then $\tau_i \le \sigma$ and, by Lemma 7, $\iota_{A_i}(\sigma) \sqsubseteq \iota_{A_i}(\tau_i) = [\![x_i]\!]_\nu$.
- In case $M = NP$, then, from Theorem 8(2), there are $B \in \mathbb{T}^0$ and $\tau \in \Xi^\omega(B)$ such that $\Gamma \vdash^\omega_\wedge N : \tau \to \sigma$ and $\Gamma \vdash^\omega_\wedge P : \tau$. By induction $\iota_{B\to A}(\tau \to \sigma) \sqsubseteq [\![N]\!]_\nu$ and $\iota_B(\tau) \sqsubseteq [\![P]\!]_\nu$, thus, $\iota_A(\sigma) = \iota_{B\to A}(\tau \to \sigma)(\iota_B(\tau)) \sqsubseteq [\![N]\!]_\nu(\iota_B(\tau))$ and, by monotonicity, $[\![N]\!]_\nu(\iota_B(\tau)) \sqsubseteq [\![N]\!]_\nu([\![P]\!]_\nu) = [\![NP]\!]_\nu$. From this we finally get $\iota_A(\sigma) \sqsubseteq [\![NP]\!]_\nu$.
- In case $M = \lambda x.N$, then by Theorem 8(3) we have that $A = B \to C$ and, for all $j \in [1;n]$, there are $\sigma_j \in \Xi^\omega(B)$, $\sigma'_j \in \Xi^\omega(C)$ such that $\sigma = \bigwedge_{j=1}^n \sigma_j \to \sigma'_j$ and $\Gamma, x : \sigma_j \vdash^\omega_\wedge N : \sigma'_j$. Thus, by induction hypothesis, we get $\iota_C(\sigma'_j) \sqsubseteq [\![N]\!]_{\nu[x:=\iota_B(\sigma_j)]}$. From Lemma 5(ii) it ensues that $\iota_A(\sigma) \sqsubseteq [\![M]\!]_\nu$.

($2 \Rightarrow 1$) It suffices to establish by induction that $[\![M]\!]_\nu = \iota_A(\sigma)$, for all ν such that $\nu(x_i) = \iota_{A_i}(\tau_i)$, entails $\Gamma \vdash^\omega_\wedge M : \sigma$. Indeed, if τ is such that $\iota_A(\tau) \sqsubseteq [\![M]\!]_\nu$ then by Lemma 7 and $\sigma \le \tau$ we obtain, using the subsumption rule, that $\Gamma \vdash^\omega_\wedge M : \tau$.

- If $M = x_i$, then $[\![x_i]\!]_\nu = \iota_{A_i}(\tau_i) = \iota_A(\sigma)$ and $\sigma \simeq \tau_i$. Thus $\Gamma \vdash^\omega_\wedge x_i : \sigma$.
- If $M = NP$, then there is B such that $\Delta \vdash N : B \to A$ and $\Delta \vdash P : B$. By Corollary 2, there are $\tau \in \Xi^\omega(B \to A), \rho \in \Xi^\omega(B)$ such that $[\![N]\!]_\nu = \iota_{B\to A}(\tau)$ and $[\![P]\!]_\nu = \iota_B(\rho)$. The induction hypothesis implies that $\Gamma \vdash^\omega_\wedge N : \tau$ and $\Gamma \vdash^\omega_\wedge P : \rho$ are derivable. By hypothesis we know that $[\![M]\!]_\nu = \iota_A(\sigma)$. From Lemma 5(ii), since $\iota_A(\sigma) = [\![M]\!]_\nu = [\![N]\!]_\nu([\![P]\!]_\nu) = \iota_{B\to A}(\tau)(\iota_B(\rho))$, we have $\iota_B(\rho) \mapsto \iota_A(\sigma) \sqsubseteq \iota_{B\to A}(\tau)$ and thus, by Lemma 7, $\tau \le \rho \to \sigma$. Hence $\Gamma \vdash^\omega_\wedge N : \rho \to \sigma$ is derivable, which implies that $\Gamma \vdash^\omega_\wedge M : \sigma$ is derivable.

– If $M = \lambda x.N$, then $A = B \to C$. By Corollary 2 we can choose, for every $g \in \mathcal{S}_B$, $\sigma_g \in \Xi^\omega(B)$ such that $\iota_B(\sigma_g) = g$ and $\tau_g \in \Xi^\omega(C)$ such that $\iota_C(\tau_g) = [\![N]\!]_{\nu[x:=g]} = [\![M]\!]_\nu(g)$. By the induction hypothesis, for every $g \in \mathcal{S}_B$, we have $\Gamma, x : \sigma_g \vdash^\omega_\wedge N : \tau_g$. Therefore, $\Gamma \vdash^\omega_\wedge M : \sigma_g \to \tau_g$ and $\Gamma \vdash^\omega_\wedge M : \bigwedge_{g\in\mathcal{S}_B} \sigma_g \to \tau_g$. By definition $\iota_A(\bigwedge_{g\in\mathcal{S}_B} \sigma_g \to \tau_g) = \bigsqcup_{g\in\mathcal{S}_b} \iota_B(\sigma_g) \mapsto \iota_C(\tau_g)$ $= \bigsqcup_{g\in\mathcal{S}_b} g \mapsto [\![M]\!]_\nu(g)$ which is equal, by Lemma 6, to $[\![M]\!]_\nu$. □

4 Inhabitation Reduces to Definability

We now prove that the undecidability of the Definability Problem follows from the undecidability of the inhabitation problem (for game types) in CDV. A preliminary version of this result was announced in the invited paper [8].

The proof we present here is obtained by linking via a suitable logical relation $\mathscr{I}$ the continuous model $\mathcal{S}$ built in the previous section and $\mathcal{F} = \{\mathcal{F}_A\}_{A\in\mathbb{T}^0} = \mathrm{Full}(\mathcal{P}(X))$, where $X \subseteq_{\mathrm{f}} \mathbb{A}$. Let $\mathscr{I}$ be the logical relation between $\mathcal{S}$ and $\mathcal{F}$ generated by taking the identity at ground level (indeed $\mathcal{S}_0 = \mathcal{F}_0 = \mathcal{P}(X)$).

Lemma 8. *$\mathscr{I}$ is a* logical retract*, i.e. at every level $A \in \mathbb{T}^0$ we have $\forall f_1, f_2 \in \mathcal{S}_A$, $\mathscr{I}_A(f_1) \cap \mathscr{I}_A(f_2) \neq \emptyset$ iff $f_1 = f_2$. Equivalently, both next statements hold:*

(i) for all $f \in \mathcal{S}_A$ there is $g \in \mathcal{F}_A$ such that $f \ \mathscr{I}_A \ g$,
(ii) for all $f, f' \in \mathcal{S}_A, g \in \mathcal{F}_A$ if $f \ \mathscr{I}_A \ g$ and $f' \ \mathscr{I}_A \ g$ then $f = f'$.

Proof. We prove the main statement by induction on A, then both items follow. The base case $A = 0$ is trivial, so we consider the case $A = B \to C$.

($\Rightarrow$) By definition of $\mathscr{I}_A(f_1), \mathscr{I}_A(f_2)$ we have:

$$\mathscr{I}_A(f_1) \cap \mathscr{I}_A(f_2) = \{h \mid \forall g \in \mathcal{S}_B, \forall k \in \mathscr{I}_B(g), h(k) \in \mathscr{I}_C(f_1(g)) \cap \mathscr{I}_C(f_2(g))\}.$$

Now, $\mathscr{I}_A(f_1) \cap \mathscr{I}_A(f_2) \neq \emptyset$ entails $\mathscr{I}_C(f_1(g)) \cap \mathscr{I}_C(f_2(g)) \neq \emptyset$ for all $g \in \mathcal{S}_B$. By induction, this holds when $f_1(g) = f_2(g)$ for all $g \in \mathcal{S}_B$, i.e. when $f_1 = f_2$.

($\Leftarrow$) If $f_1 = f_2$ then $\mathscr{I}_A(f_1) = \{h \mid \forall g \in \mathcal{S}_B, \forall k \in \mathscr{I}_B(g), h(k) \in \mathscr{I}_C(f_1(g))\}$. To prove $\mathscr{I}_A(f_1) \neq \emptyset$, we build a relation $h \subseteq \mathcal{F}_B \times \mathcal{F}_C$ that is actually functional and belongs to it. Fix any $d \in \mathcal{F}_C$ and, for every $g \in \mathcal{S}_B$, an element $r_g \in \mathscr{I}_C(f_1(g))$ which exists by induction hypothesis. Define h as the smallest relation such that $(k, r_g) \in h$ if $k \in \mathscr{I}_B(g)$, and $(k, d) \in h$ if $k \notin \bigcup_{g\in\mathcal{S}_B} \mathscr{I}_B(g)$. As, by induction hypothesis, $\mathscr{I}_B(g_1)$ and $\mathscr{I}_B(g_2)$ are disjoint for all $g_1 \neq g_2$ then h is functional. By construction, $h \in \mathscr{I}_C(f_1(g))$. □

As a consequence we get, for every subset $S \subseteq \mathcal{S}_A$, that $\mathscr{I}_A^-(\mathscr{I}_A(S)) = S$. Given $f \in \mathcal{S}_A$ we write $f\!\uparrow$ for its upward closure in $\mathcal{S}_A$: $\{f' \in \mathcal{S}_A \mid f \sqsubseteq f'\}$.

Proposition 2. *Let $\sigma \in \Xi(A)$. For every normal λ-term M having type A we have $\vdash_\wedge M : \sigma$ iff $[\![M]\!]^{\mathcal{F}} \in \mathscr{I}_A(\iota_A(\sigma)\!\uparrow)$.*

Proof. We have the following computable chain of equivalences:

$$\begin{aligned} \vdash_\wedge M : \sigma &\iff \vdash^\omega_\wedge M : \sigma, && \text{by Lemma 4,} \\ &\iff [\![M]\!]^{\mathcal{S}} \in \iota_A(\sigma)\!\uparrow, && \text{by Proposition 1,} \\ &\iff [\![M]\!]^{\mathcal{F}} \in \mathscr{I}_A(\iota_A(\sigma)\!\uparrow), && \text{by Lemma 1 plus Lemma 8.} \end{aligned}$$ □

Theorem 9. *The undecidability of the Definability Problem follows by a reduction from the one of the Inhabitation Problem for game types, Theorem 2(2).*

Proof. Suppose by contradiction that DP is decidable. We want to decide whether $\sigma \in \bigcup_{A\in\mathbb{T}^0, X\subseteq_f \mathbb{A}} \Xi_X(A)$ is inhabited in CDV. By Theorem 3 and Corollary 1 we can focus on normal simply typed λ-terms. Now we can take the set Y of all atoms in σ, compute the simple type A such that $\sigma \in \Xi_Y(A)$, and effectively construct the finite set $\mathscr{I}_A(\iota_A(\sigma)\!\uparrow) \subseteq \mathrm{Full}(Y)$. If DP is decidable, then we can also decide with finitely many tests whether there is a λ-definable $f \in \mathscr{I}_A(\iota_A(\sigma)\!\uparrow)$. By Proposition 2 such an f exists if and only if σ is inhabited. This yields a reduction of IHP for game types (hence for uniform types, Theorem 6) to DP. □

5 Definability Reduces to Inhabitation

In this section we prove the converse of Theorem 9, namely that the undecidability of inhabitation follows directly from the undecidability of λ-definability in the full model $\mathcal{F} = \mathrm{Full}(X)$ over a fixed set $X \subseteq_f \mathbb{A}$. The main idea is a simple embedding of the elements of $\mathcal{F}$ into the uniform intersection types.

Also in this proof the continuous model $\mathcal{S} = \mathrm{Cont}(\mathcal{P}(X), \subseteq)$ will play a key role. (Remark that the ground set of $\mathcal{S}$ is still $\mathcal{P}(X)$, while $\mathcal{F}$ is now over X.) We start by defining an injection $\varphi_A : \mathcal{F}_A \to \mathcal{S}_A$ by induction on A:

- if $A = 0$, then $\varphi_A(f) = \{f\}$,
- if $A = B \to C$, then $\varphi_A(f) = \bigsqcup_{g\in\mathcal{F}_B} \varphi_B(g) \mapsto \varphi_C(f(g))$.

Now, given f in $\mathcal{F}_A$ we define an intersection type ξ_f in $\Xi(A)$ as follows:

- if $A = 0$, then $\xi_f = f$,
- if $A = B \to C$, then $\xi_f = \bigwedge_{g\in\mathcal{F}_B} \xi_g \to \xi_{f(g)}$.

Lemma 9. *For every f in $\mathcal{F}_A$, we have $\varphi_A(f) = \iota_A(\xi_f)$.*

We consider the logical relation $\mathscr{J}$ between the full model $\mathcal{F}$ and the continuous model $\mathcal{S}$ generated by $\mathscr{J}_0 = \{(f, F) \mid f \in F \subseteq \mathcal{F}_0\}$.

Lemma 10. *For every $f \in \mathcal{F}_A$ and $g \in \mathcal{S}_A$ we have $f\ \mathscr{J}_A\ g$ iff $\varphi_A(f) \sqsubseteq g$.*

Proof. By induction on A, the case $A = 0$ being obvious. Let $A = B \to C$.

($\Rightarrow$) Suppose $f\ \mathscr{J}_A\ g$. We want to prove that $\varphi_A(f) \sqsubseteq g$. That is, for all $h \in \mathcal{S}_B$, we have $\varphi_A(f)(h) \sqsubseteq g(h)$. Let $h \in \mathcal{S}_B$, then by definition of φ_A, we have $\varphi_A(f)(h) = \bigsqcup\{\varphi_C(f(k)) \mid \varphi_B(k) \sqsubseteq h, k \in \mathcal{F}_B\}$. But $\varphi_B(k) \sqsubseteq h$ implies $k\ \mathscr{J}_B\ h$ by induction hypothesis, which implies that $f(k)\ \mathscr{J}_C\ g(h)$ since $f\ \mathscr{J}_A\ g$. Now using the induction hypothesis for C, we get $\varphi_C(f(k)) \sqsubseteq g(h)$. That is, $\varphi_A(f)(h)$ is a supremum of things all of which are below $g(h)$, thus $\varphi_A(f)(h) \sqsubseteq g(h)$.

($\Leftarrow$) Suppose $\varphi_A(f) \sqsubseteq g$. Let $h \in \mathcal{F}_B$ and $h' \in \mathcal{S}_B$ with $h\ \mathscr{J}_B\ h'$, that is, by the induction hypothesis, with $\varphi_B(h) \sqsubseteq h'$. We want to show that $f(h)\ \mathscr{J}_C\ g(h')$ or, equivalently, again by the induction hypothesis, that $\varphi_C(f(h)) \sqsubseteq g(h')$. Now, by definition, $\varphi_A(f)(h') = \bigsqcup\{\varphi_C(f(k)) \mid \varphi_B(k) \sqsubseteq h', k \in \mathcal{F}_B\}$, and by assumption $h \in \mathcal{F}_B$ and $\varphi_B(h) \sqsubseteq h'$, so $\varphi_C(f(h)) \sqsubseteq \varphi_A(f)(h')$. On the other hand, $\varphi_A(f) \sqsubseteq g$ as functions on $\mathcal{S}_A$ and $h' \in \mathcal{S}_B$, so $\varphi_A(f)(h') \sqsubseteq g(h')$. By transitivity of the order we obtain $\varphi_C(f(h)) \sqsubseteq g(h')$ as required. □

Proposition 3. *Given f in $\mathcal{F}_A$, we have $[\![M]\!]^{\mathcal{F}} = f$ iff $\vdash_\wedge M : \xi_f$.*

Proof. We have the following computable chain of equivalences:

$$
\begin{aligned}
[\![M]\!]^{\mathcal{F}} = f &\iff f \mathrel{\mathscr{J}_A} [\![M]\!]^{\mathcal{S}}, && \text{by Lemma 1,}\\
&\iff \varphi(f) \sqsubseteq [\![M]\!]^{\mathcal{S}}, && \text{by Lemma 10,}\\
&\iff \iota_A(\xi_f) \sqsubseteq [\![M]\!]^{\mathcal{S}}, && \text{by Lemma 9,}\\
&\iff \vdash_\wedge M : \xi_f, && \text{by Proposition 1.}
\end{aligned}
$$
□

Therefore f is definable iff ξ_f is inhabited. This yields a reduction of the Definability Problem (resp. DP_n) to the Inhabitation Problem (resp. IHP_n).

Theorem 10. *1. The undecidability of IHP_n for all $n > 1$ follows by a reduction from the undecidability of DP_n for all $n > 1$, Theorem 1(2).*

2. The undecidability of the Inhabitation Problem follows by a reduction from the undecidability of the Definability Problem, Theorem 1(1).

Acknowledgements. We are grateful to Antonio Bucciarelli for interesting discussions. This work is partly supported by NWO Project 612.000.936 CALMOC, ANR 2010 BLAN 0202 01 FREC and ANR 2010 BLAN 0202 02 FREC.

References

1. S. Abramsky. *Domain theory in logical form.* In Symposium on Logic and Computer Science (LICS'87), IEEE Computer Science Press, pp. 47-53, 1987.
2. R. Amadio and P.-L. Curien. *Domains and lambda-calculi.* Cambridge Tracts in Theoretical Computer Science, no. 46, Cambridge University Press, 1998.
3. H.P. Barendregt, W. Dekkers and R. Statman. *Lambda calculus with types.* To appear. Draft available at http://www.cs.ru.nl/∼henk/book.pdf.
4. M. Coppo, M. Dezani-Ciancaglini, and B. Venneri. *Functional characters of solvable terms.* Mathematical Logic Quarterly, Volume 27, Issue 2-6, pages 45-58, 1981.
5. T. Joly. *Encoding of the halting problem into the monster type & applications.* Typed Lambda Calculi and Applications (TLCA'03), LNCS, vol. 2701, pp. 153-166, 2003.
6. R. Loader. *The undecidability of lambda definability.* In Logic, Meaning and Computation: Essays in Memory of Alonzo Church, 331-342, 2001.
7. G. Plotkin. *Lambda definability and logical relations.* Memorandum SAI-RM-4, School of Artificial Intelligence, University of Edinburgh, 1973.
8. S. Salvati. *Recognizability in the simply typed lambda-calculus.* Logic, Language, Information and Computation (WoLLIC'09), LNCS, vol. 5514, pp. 48-60, 2009.
9. R. Statman. *Completeness, invariance and λ-definability.* The Journal of Symbolic Logic, vol. 47, no. 1, pp. 17-26, 1982.
10. P. Urzyczyn. *The emptiness problem for intersection types.* The Journal of Symbolic Logic, vol. 64, no. 3, pp. 1195-1215, 1999.